Improved Protein-Ligand Prediction Using Kernel Weighted Canonical Correlation Analysis

Raissa Relator[1,a)] Tsuyoshi Kato[1,b)] Richard Lemence[2]

Abstract: Protein-ligand interaction prediction plays an important role in drug design and discovery. However, wet lab procedures are inherently time consuming and expensive due to the vast number of candidate compounds and target genes. Hence, computational approaches became imperative and have become popular due to their promising results and practicality. Such methods require high accuracy and precision outputs for them to be useful, thus, the problem of devising such an algorithm remains very challenging. In this paper we propose an algorithm employing both support vector machines (SVM) and an extension of canonical correlation analysis (CCA). Following assumptions of recent chemogenomic approaches, we explore the effects of incorporating bias on similarity of compounds. We introduce kernel weighted CCA as a means of uncovering any underlying relationship between similarity of ligands and known ligands of target proteins. Experimental results indicate statistically significant improvement in the area under the ROC curve (AUC) and F-measure values obtained as opposed to those gathered when only SVM, or SVM with kernel CCA is employed, which translates to better quality of prediction.

Keywords: canonical correlation analysis, kernel methods, protein-ligand interaction, support vector machines

1. Introduction

Drug discovery is a multi-staged process which involves the determination of existing interactions between a compound and a protein. Many drugs are developed depending on the reaction they produce when coupled with the respective proteins acting during a biological process in the body. However, only a few existing interactions have actually been validated through experiments. Moreover, wet lab procedures are inherently time consuming and expensive due to the vast number of candidate compounds and target genes. Hence, computational approaches became imperative and have become popular due to their promising results and practicality.

The protein-ligand interaction prediction problem can be viewed as a task of filling up a protein-ligand matrix whose rows represent the candidate compounds and the columns represent the target proteins as shown in the example in Figure 1(a). A matrix entry is $+1$ if there is interaction between the corresponding drug and target. Otherwise, -1. Only a few interactions have actually been verified and recorded which makes the protein-ligand matrix sparse. Termed as the ‘chemogenomic approach’ by Rognan [13], the ultimate goal of this task is to identify all the ligands of each target, thus, fully matching the ligand and target spaces [1].

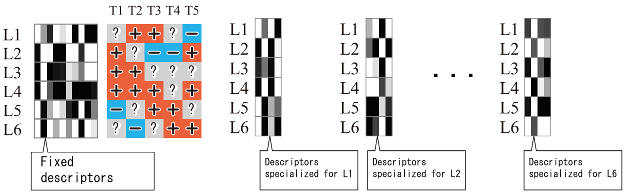
Many in silico methods have already been developed to address this problem. We can classify these methods into two: the structure or docking approach and the ligand-based approach. Docking approaches make use of 3D structures of the chemical compounds or the proteins to find protein-ligand pairs which are more likely to bind [2], [3]. On the other hand, ligand-based techniques usually employ machine learning algorithms in comparing known ligands and candidate ligands of a certain target even without any prior information regarding their structure [7], [8]. In this study, we shall make use of the ligand-based approach.

There are two ways of approaching the task of interaction prediction: one is by using the global model [11], and another one is via the local model [1], [8]. The global model utilizes a large interaction matrix and imputation of missing values is done simultaneously. Each cell in the interaction matrix is considered as a sample to which statistical methods are applied. Descriptors of ligands in the form of a feature matrix and some information for target proteins are combined to generate a fused profile for each cell in the interaction matrix. An advantage is that interaction prediction for target proteins with few known interactions can still be formed. However, since the model aims to exploit information from similar columns, some useful information for learning the rule for prediction may be corrupted by information from irrelevant columns.

Meanwhile, in the local model approach, prediction is made for each column of the protein-ligand table independently — the approach finds unknown chemical compounds which are similar to known ligands interacting with the target protein of interest. The local model often suffers from a small-sample problem. Many columns in the protein-ligand interaction matrix include few positive interactions, causing machine learning algorithms to be trained with few positive samples despite very high dimensionality of ligand descriptors.

The goal of determining interactions between targets and com-

[1] Department of Computer Science, Graduate School of Science and Engineering, Gunma University
[2] Institute of Mathematics, College of Science, University of the Philippines, Diliman
[a)] relator-raissa@kato-lab.cs.gunma-u.ac.jp
[b)] katotsu@cs.gunma-u.ac.jp



(a) Problem (b) Specialized descriptors

Fig. 1 Protein-ligand matrix and descriptors. In the example depicted in (a), the prediction task is to impute 11 missing entries in the 6×5 protein-ligand matrix using 10-dimensional raw descriptors of ligands. The problem can be divided into six sub-problems, each of which is to complete a row in the protein-ligand matrix. Our algorithm extracts compact descriptors specialized in each sub-problem.

pounds is established under twofold assumptions [1], [13]: First is that compounds with similar properties tend to share targets. And, targets with similar ligands share similarities in structures such as binding sites. These have been verified by recent studies by considering drug side effects [5] and similarities among ligands [10]. Moreover, integrated approaches exploring both protein and compound similarities have also been investigated [4], [8]. Thus, recent methodologies have allowed us to make predictions on interactions based on similarity measures for ligands and targets.

Motivated by the assumption that similar ligands tend to have similar target proteins [9], [15], our goal is to uncover any underlying relationship between a set of ligands and exploit this relationship, together with some known ligand-target interactions, to predict new interactions. We search for ligands with strong associations by finding correlations between them using their features.

In this paper, we present a weighted extension of canonical correlation analysis (WCCA) in the reproducing kernel Hilbert space (RKHS) in an attempt to introduce advantageous properties of local models to the global model approach. To estimate the missing entries in each row of the interaction matrix, we use kernel WCCA (KWCCA) to extract essential features which are specialized in imputation of the corresponding row. The extracted features are compact enough for local models to be trained with a small training set composed from the column. Through the experiments with data of GPCRs and odorant receptors, the prediction performance is shown to be improved when our algorithm is applied compared to several existing methods.

2. Materials and Methods

2.1 Data

The data used for this study was originally from [14]. The given interaction matrix consists of 62 mammalian odorant receptors (ORs) as target proteins and 63 odorants as candidate ligands. It is binary in form and contains 340 positive interactions. The number of known positive interactions for each target protein is at least one and at most thirty-seven, while the median is three. Some randomly selected protein-ligand pairs are assumed to be unknown to test prediction methods, and the values of the cells are set to zero. Each row in the interaction matrix provides an *interaction profile* of the ligand.

From the chemical IDs supplied, we searched PubChem[*1] for the chemical structures of the odorants to obtain the descriptors of the ligands. Frequent substructures are employed as descriptors of ligands. The frequent substructures are mined with a software named *gSpan* [18]. The software is applied to the 63 chemical structures, and the $60,311$ binary descriptors are obtained as *chemical profiles*.

2.2 Overview of the algorithm

Our approach consists of two stages: First, we consider subproblems, each of which involves imputation on a single row in the interaction matrix, and use weighted CCA to extract a compact vector representation for each sub-problem. Then, we apply SVM for prediction of each cell using the corresponding descriptor extracted in the previous stage. This technique is overviewed as follows.

Chemical profiles obtained from chemical structures contain numerous features that are not important for prediction. Extracting significant features from such chemical profiles is crucial for accurate prediction of protein-ligand interaction. To accomplish this, we have to find effective low-dimensional representations of the original chemical profiles lying in the extremely high-dimensional *chemical space*.

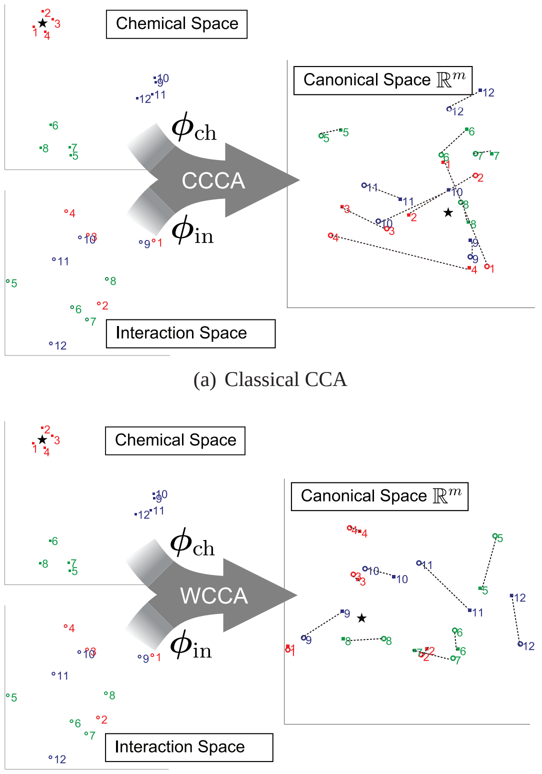
Interaction profiles describe the existence and the absence of interactions with several target proteins. More often than not, target proteins share similar properties. For this reason, interaction profiles approximately span a low-dimensional space, say $\mathbb{R}^m$, which we shall also extract from a high-dimensional *interaction space*, in a similar fashion as the chemical profiles.

Canonical correlation analysis uses a set of chemical profiles and interaction profiles to find two projection functions, ϕ_{ch} and ϕ_{in}, simultaneously: The projection ϕ_{ch} is from the chemical space to the low-dimensional canonical space $\mathbb{R}^m$, and ϕ_{in} is from the interaction space to $\mathbb{R}^m$. The images of ϕ_{ch} are used to approximate the images of ϕ_{in}. The projections obtained by CCA are shown mathematically to be the minimizer of the expected deviation of the image of ϕ_{ch} from the image of ϕ_{in}.

Figure 2(a) is an illustration of how CCA works with chemical profiles and interaction profiles. In this figure, the shaded squares are data representations of the feature vector of each ligand in the chemical space. While the open circles are the data representation of the interaction vector of each ligand in the interaction space. The images under ϕ_{ch} and ϕ_{in} of these data points are plotted in the canonical space, and their corresponding images are linked with a dashed line. CCA finds the projections ϕ_{ch} and ϕ_{in} so that the average squared length of the dashed lines is minimized.

In application to protein-ligand interaction prediction, estimating the images for all ligands is not necessary; it is only for the ligand whose interactions we wish to predict that the image of the chemical compound is desired to be well approximated. To obtain a good approximation for a ligand of interest, it is sufficient to estimate projections so that only the images of similar ligands are approximated well. The precisions of the approximations for ligands dissimilar to the ligand of interest barely affect the accu-

[*1] http://pubchem.ncbi.nlm.nih.gov/



(a) Classical CCA

(b) Weighted CCA

Fig. 2 Classical CCA and weighted CCA. Our approach projects chemical and interaction profiles into a low-dimensional canonical space so that the images are close to each other. The star point represents the ligand of interest, and red points are ligands sharing similarities with the ligand of interest. Although the classical CCA minimizes the average deviation over all the ligands, to achieve accurate prediction, it is sufficient that the deviations between the images of the target ligand and the ligands similar to it are small. The weighted CCA works with arbitrarily specified weights, which ensures small deviations for red points by giving them larger weights.

racy of the solution. This consideration motivated us to assign weights to ligands according to their similarity to the ligand of interest, and to extend the classical CCA so that the weighted average deviation is minimized. The weighted CCA almost disregards ligands with small weights to find projections, achieving more accurate approximations for the ligand of interest. We refer to the extension of CCA as weighted CCA.

Figure 2(b) illustrates the effects of weighted CCA when weights are added to similar ligands. In this context, we define similarity as the measure of affinity between features of compounds. This can be represented by the distance between the data representation of the ligands in the chemical space. In the given figure, the chemical profile for a ligand of interest is marked with a star, and profiles of similar ligands are colored red. In a similar manner, we interpret points of the same color as ligands sharing similarities in their chemical properties, hence their grouping in the chemical space. The two figures, (a) and (b), allow us to compare classical CCA with weighted CCA: the deviations for red points in (b) are smaller than those in (a). The deviations for other ligands are larger, which hardly worsen the performance of predicting the interaction of the protein of interest.

The final prediction result is obtained in the post-processing stage using SVM. The images of the projections are used for SVM learning. SVM is trained well if a good training set is given. Hence, ligands with poor approximations by CCA, which are noisy for SVM learning, are preferably excluded. The images are already in a low-dimensional space in which SVM learning works well even with a small training set, encouraging us to assign smaller weights to ligands with poor approximations for SVM learning.

2.3 Weighted CCA

In this subsection we present the details of weighted CCA. We denote the chemical profile and the interaction profile, respectively, by a p_{ch}-dimensional vector $\boldsymbol{x}^{\text{ch}}$ and a p_{in}-dimensional vector $\boldsymbol{x}^{\text{in}}$. Assuming that the functions $\phi_{\text{ch}} : \mathbb{R}^{p_{\text{ch}}} \to \mathbb{R}^m$ and $\phi_{\text{in}} : \mathbb{R}^{p_{\text{in}}} \to \mathbb{R}^m$ are affine transformations allows us to express them as

$$\phi_{\text{ch}}(\boldsymbol{x}^{\text{ch}}) = \boldsymbol{W}_{\text{ch}}^{\top}(\boldsymbol{x}^{\text{ch}} - \boldsymbol{\mu}_{\text{ch}}), \quad \phi_{\text{in}}(\boldsymbol{x}^{\text{in}}) = \boldsymbol{W}_{\text{in}}^{\top}(\boldsymbol{x}^{\text{in}} - \boldsymbol{\mu}_{\text{in}}),$$

where $\boldsymbol{W}_{\text{ch}} \in \mathbb{R}^{p_{\text{ch}} \times m}$, $\boldsymbol{\mu}_{\text{ch}} \in \mathbb{R}^{p_{\text{ch}}}$, $\boldsymbol{W}_{\text{in}} \in \mathbb{R}^{p_{\text{in}} \times m}$, and $\boldsymbol{\mu}_{\text{in}} \in \mathbb{R}^{p_{\text{in}}}$ are their respective parameters. We wish to find the pair of projection functions minimizing the expected deviation between the images given by

$$J(\phi_{\text{ch}}, \phi_{\text{in}}) \equiv \mathbb{E}\left[\left\|\phi_{\text{ch}}(\boldsymbol{x}^{\text{ch}}) - \phi_{\text{in}}(\boldsymbol{x}^{\text{in}})\right\|^2\right],$$

where $\mathbb{E}$ is the expectation operator.

The expected deviation can be reduced arbitrarily by setting the projections so that the images are scaled down. A trivial solution is $\boldsymbol{W}_{\text{ch}} = \boldsymbol{0}$ and $\boldsymbol{W}_{\text{in}} = \boldsymbol{0}$ at which the expected deviation vanishes for any dataset. To avoid trivial solutions, the size of the images is adjusted by fixing the second moment matrices, $\mathbb{E}[\phi_{\text{ch}}(\boldsymbol{x}^{\text{ch}})\phi_{\text{ch}}(\boldsymbol{x}^{\text{ch}})^{\top}]$ and $\mathbb{E}[\phi_{\text{in}}(\boldsymbol{x}^{\text{in}})\phi_{\text{in}}(\boldsymbol{x}^{\text{in}})^{\top}]$, to identity matrices.

The expectation appearing in the derivation and the second moment matrices operates according to an empirical probabilistic distribution. Supposing n ligands are given, the chemical profiles are denoted by $\boldsymbol{x}_1^{\text{ch}}, \ldots, \boldsymbol{x}_n^{\text{ch}}$, and the interaction profiles by $\boldsymbol{x}_1^{\text{in}}, \ldots, \boldsymbol{x}_n^{\text{in}}$. If we define an empirical distribution as

$$q(\boldsymbol{x}^{\text{ch}}, \boldsymbol{x}^{\text{in}}) = \sum_{j=1}^{n} v_j \delta(\boldsymbol{x}^{\text{ch}} - \boldsymbol{x}_j^{\text{ch}})\delta(\boldsymbol{x}^{\text{in}} - \boldsymbol{x}_j^{\text{in}}),$$

with weights $v_1, \ldots, v_n$ whose sum is one and $\delta(\cdot)$ is the Dirac delta function, then the expected deviation is reduced to the weighted average of deviation and can be expressed as

$$J(\phi_{\text{ch}}, \phi_{\text{in}}) = \sum_{j=1}^{n} v_j \left\|\phi_{\text{ch}}(\boldsymbol{x}_j^{\text{ch}}) - \phi_{\text{in}}(\boldsymbol{x}_j^{\text{in}})\right\|^2. \qquad (1)$$

This implies that approximations are refined locally by setting the weights so that ligands dissimilar from the target ligand are given smaller weights.

The optimal projections can be computed via the generalized eigen-decomposition. When setting $v_j = 1/n$, the algorithm is shown to be equivalent to the classical CCA. Hence, we can say that weighted CCA is an extension of the classical CCA.

Kernelization of weighted CCA is formulated with a similarity function of chemical profiles $K_{\text{ch}}(\boldsymbol{x}_i^{\text{ch}}, \boldsymbol{x}_j^{\text{ch}})$ and a similarity function of interaction profiles $K_{\text{in}}(\boldsymbol{x}_i^{\text{in}}, \boldsymbol{x}_j^{\text{in}})$ without using the vectors themselves explicitly. These similarity functions are said to be valid kernels guaranteeing the theory of the algorithms, which map the profiles non-linearly into other (typically high-dimensional) spaces $\mathcal{H}_{\text{ch}}$ and $\mathcal{H}_{\text{in}}$, respectively, called an RKHS. Kernelized weighted CCA finds affine-transforms from RKHS to a canonical space $\mathbb{R}^m$, so that the expected deviation between images in $\mathbb{R}^m$ is minimized. If we denote the composite mapping functions by ψ_{ch} and ψ_{in}, respectively, the optimal solution is given by

$$\psi_{\text{ch}}(\boldsymbol{x}^{\text{ch}}) = \boldsymbol{A}_{\text{ch}}^\top \boldsymbol{D}_{\boldsymbol{v}}^{1/2} \bar{\boldsymbol{k}}_{\text{ch}}(\boldsymbol{x}^{\text{ch}}), \quad \psi_{\text{in}}(\boldsymbol{x}^{\text{in}}) = \boldsymbol{A}_{\text{in}}^\top \boldsymbol{D}_{\boldsymbol{v}}^{1/2} \bar{\boldsymbol{k}}_{\text{in}}(\boldsymbol{x}^{\text{in}}).$$

The algorithm for computing the two matrices, $\boldsymbol{A}_{\text{ch}} \in \mathbb{R}^{m\times n}$ and $\boldsymbol{A}_{\text{in}} \in \mathbb{R}^{m\times n}$. The functions $\bar{\boldsymbol{k}}_{\text{ch}}(\cdot)$ and $\bar{\boldsymbol{k}}_{\text{in}}(\cdot)$ are called the empirical kernel mapping.

2.4 Weighted SVM

Prediction of the interaction between ligand i and target t is performed with the SVM score given by

$$f(\boldsymbol{x}_i^{\text{ch}};\, \boldsymbol{w}_{(i,t)}, b_{(i,t)}) = \boldsymbol{w}_{(i,t)}^\top \psi_{\text{ch}}(\boldsymbol{x}_i^{\text{ch}}) + b_{(i,t)},$$

where $\boldsymbol{x}_i^{\text{ch}}$ is the chemical profile of ligand i. The SVM parameters, $\boldsymbol{w}_{(i,t)}$ and $b_{(i,t)}$, are obtained beforehand by the SVM learning algorithm. This is performed only with ligands whose interaction with the target t is known. This study employs the similarity of ligands as weights in the learning process.

2.5 Weighting schemes

Ligands are given weights in both stages of the weighted CCA and the weighted SVM. These weights are dependent on the ligand to be predicted. Larger weights are given for ligands that are more similar to the ligand of interest. In predicting the interaction of the ith ligand, the weight of jth ligand is given by the normalization of

$$v'_j = \frac{1}{\left\|\bar{\boldsymbol{k}}_{\text{ch}}(\boldsymbol{x}_j^{\text{ch}}) - \bar{\boldsymbol{k}}_{\text{ch}}(\boldsymbol{x}_i^{\text{ch}})\right\| + \left\|\bar{\boldsymbol{k}}_{\text{in}}(\boldsymbol{x}_j^{\text{in}}) - \bar{\boldsymbol{k}}_{\text{in}}(\boldsymbol{x}_i^{\text{in}})\right\| + \epsilon},$$

where ϵ is a positive constant and set to 10 in our analysis. Normalization is done by setting

$$v_j = \frac{v'_j}{\sum_{k=1}^n v'_k}$$

so that the sum of the weights is one.

3. Results

3.1 Experimental setting

To illustrate the effectiveness of the kernel weighted CCA (KWCCA), we carried out experiments on an interaction dataset of GPCRs and odorant receptors described in the previous section. For evaluation of prediction performance, we applied a 10-fold Monte-Carlo cross validation, where data is randomly divided into 2 disjoint sets of training and test data for 10 repetitions. Data was partitioned such that for each target protein, 50% of the positive and negative interactions are used for training, and the other half for testing. KCCA, KWCCA, and the weighted SVM were implemented in Matlab, and LIBSVM [6] was used for the classical SVM.

Table 1 Abbreviation of methods.

Abbreviation	Description
WW	KWCCA + Weighted SVM
WU	KWCCA + Classical SVM
KW	Classical KCCA + Weighted SVM
KU	Classical KCCA + Classical SVM
S	SVM of local model
SGL	Linear SVM of global model
SGR	RBF SVM of global model

We also performed prediction using SVM in the global model setting for comparison. The kernel function for the global model here is defined as the product of the inner product among chemical profiles and the inner product among columns of the interaction matrix.

Parameters of the local models are determined by finding respective values where the test data perform best using SVM and KCCA. Namely, the regularization parameter C and the kernel function for SVM are chosen so that SVM achieves the highest prediction performance, while the regularization parameters for CCA γ_{ch} and γ_{in}, and the number of dimensions of the canonical space m, are determined via the performance of KCCA. As a result, the values of the parameters are set as $C = 1000$, $\gamma_{\text{ch}} = \gamma_{\text{in}} = 1$, and $m = 4$. The RBF kernel is applied and the kernel width is determined as the mean of the distance within sets. These mentioned parameters are then fed into the algorithm employing KWCCA. The parameters are not tuned specifically for KWCCA. Thus, it is believed that there is a chance of improvement in the perfomance of this algorithm if careful and suitable parameter selection is done.

For the global model, the kernel which achieves the best performance is the linear kernel. The regularization parameter is chosen as $C = 10$, achieving the best performance among other values. Results for the case of the RBF kernel with the best C value obtained are also reported for comparison.

The methods based on KWCCA involve two stages upon implementation. First, we exploit KWCCA to extract a set of features for each compound. Second, we use them for training a machine learning algorithm employing SVMs before testing them to make predictions. In total, seven methods are implemented in the experiments: two using SVM in the global model setting, and the other five following the local model. One of the two global model methods uses RBF kernel for SVM, and the other uses the linear kernel. On the other hand, the methods used for the local models are as follows: SVM, KCCA with classical SVM, KCCA with weighted SVM, KWCCA with classical SVM, and KWCCA with weighted SVM. For simplicity of notation, we shall refer to each of the seven methods using the abbreviations in Table 1.

The area under the ROC curve (AUC) and F-measure values were calculated to evaluate and compare the prediction performance of the seven methods. Since the problem is presented as a binary classification problem, only the maximum value of the F-measure values for each target is considered. The scores obtained via SVM are used as confidence levels, thus, changing the threshold yields different predictions. These values are calculated for each target protein and averaged over the ten data divisions. However, there are instances when the test set does not contain a true positive interaction, hence AUC and F-measures cannot be

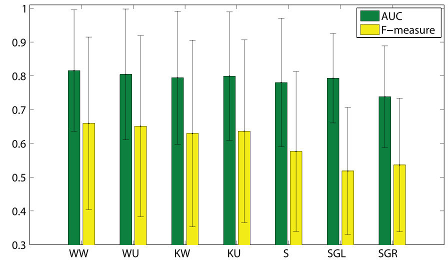


Fig. 3 Average performance of the methods. Data was randomly split into training and test sets, and 10 training-testing data divisions were used for each method. Following the local model, AUC and F-measure were computed for each of the 62 targets. The bar plots represent the average AUC (green) and average F-measure (yellow) over the 10 cross validation sets and the 49 targets containing true positives. The two KWCCA-based methods, WW and WU, and the other methods were implemented for comparison. The difference of the performances of WW and WU from the other five methods showed to be statistically significant in terms of the P-values (by Wilcoxon signed rank test).

computed. Therefore, these values were disregarded and, out of 62 target proteins, AUC and F-measures were computed for 49 of them. The Wilcoxon signed test was used for the statistical significance of the difference among the values of the evaluation measures.

3.2 Effects of the use of CCA

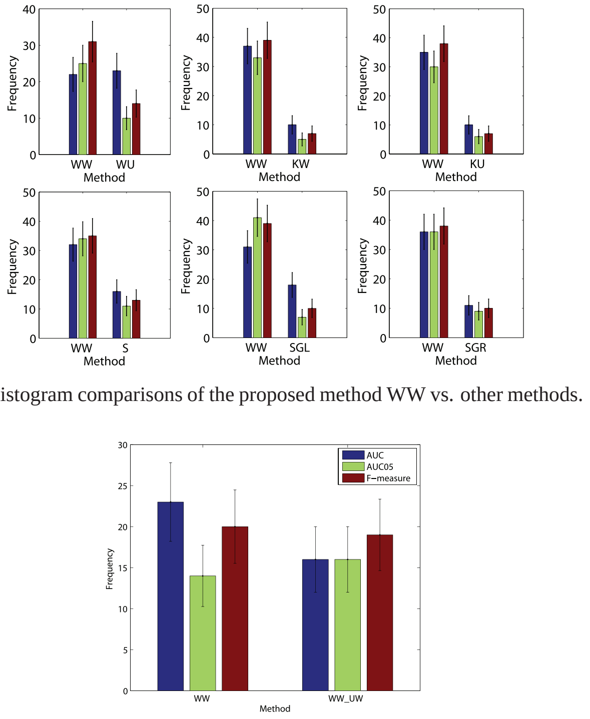
The average AUCs and F-measures are reported in Figure 3. Error bars are also included to present standard deviations. In comparison with the local models, four CCA-based methods, WW, WU, KW, and KU, achieve remarkably better AUCs and F-measures compared to those of S: the differences between the AUCs and F-measures of KW, the worst among the four CCA-based methods, and S are 0.014 and 0.053, respectively, $\left(\text{P-values: } 5.81 \times 10^{-7} \text{ and } 9.49 \times 10^{-9} \text{ respectively}\right)$. The AUC of the global model SGL is comparable to some of the local models, whereas the F-measure is not worse than that of S. A closer inspection on the results of SGL indicate that it has the lowest average number of true positives over all cross-validations among all models, around 161, which may be the reason behind a very small F-measure value.

3.3 Improvement by weighting

The effects of the weighted extension of CCA are manifested via comparison among four CCA-based methods. WW achieves significantly higher AUC and F-measures in average compared to KW and KU, where the P-values for the difference in the AUCs are 4.85×10^{-11} and 6.91×10^{-10}, and the P-values for the F-measures are 3.59×10^{-7} and 3.96×10^{-6}, respectively.

3.4 Histogram comparison

The frequencies of WW besting the AUC or F-measure values of the other methods in predicting interactions for a certain target protein are shown in the histograms in Figure 4(a). These values represent the number of target proteins such that the evaluated AUC and F-measure values for the method WW is better than the AUC and F-measure values of the other method in comparison. Instances when there are ties between the methods were unaccounted. For the evaluated AUC and F-measure values, WW outputs are more desirable than most of the others which indicates higher quality of prediction performance.



(a) Histogram comparisons of the proposed method WW vs. other methods.

(b) Histogram comparisons of Weighted SVM using the weighting scheme in (2) vs. Classical SVM

Fig. 4 Histogram comparisons between the proposed method WW and other methods. Frequencies when the AUC (blue), AUC between 0 and 0.05 (green), and F-measure (red) values of WW outperform the other methods, and vice-versa, are illustrated. It can be observed that AUC and F-measure values histograms for WW are more desirable than the rest.

WU yields interesting results in the histogram (Fig. 4(a)): The frequency of WW yielding better AUCs are comparable to that of WU's, although frequency of better F-measures are relatively higher for WW than WU. To further investigate the comparison between WW and WU, we compute the area under the curve of the region of FPR between 0 and 0.05. This area, which we shall refer to as AUC05, allows us to evaluate the true positive rate with higher confidence. The histogram on AUC05 shows WW bests WU more frequently than WU does, which implies the use of weights in the SVM stage can find more true positives confidently than the classical SVM.

The motivation to endow the weights with training data in SVM learning is that the projections in the canonical space from chemical profiles with larger weights are expected to be better approximations of the projections from interaction profiles. It is possible to directly evaluate how good the approximations are by computing the distances among the projections. This motivation leads to another weighting scheme using the normalization of

$$v'_j = \frac{1}{\left\|\phi_{\rm ch}(\boldsymbol{x}_j^{\rm ch}) - \phi_{\rm in}(\boldsymbol{x}_j^{\rm in})\right\| + \epsilon} \qquad (2)$$

instead of (1) in the SVM learning stage. We investigate the performance when the weighting scheme is changed to (2) in the SVM learning stage. We refer to this approach as WW_{UW} hereinafter. The average AUC and F-measure of $\mathrm{WW}_{\mathrm{UW}}$ are 0.802 and 0.649, respectively, which are slightly worse than those of WW. The number of target proteins, for which the prediction performance of $\mathrm{WW}_{\mathrm{UW}}$ is better than that of WW is not larger than the number of WW besting $\mathrm{WW}_{\mathrm{UW}}$, as depicted in Figure 4(b). These facts imply that the changing weighting scheme in SVM learning does not achieve significant improvements.

3.5 Using interaction profiles

When a sufficient number of known positive and negative interactions are given for a certain ligand, the image of the interaction profile in the canonical image can provide good descriptors for predicting the remaining interactions. We further implemented two methods, herein referred to as WWI and WWIC, to investigate the performance of the interaction profile. WWI replaces the image of a chemical profile with the image of the interaction profile in the SVM stage, while WWIC concatenates the two images to feed them to the weighted SVM. The two methods achieved significant improvement. WWI achieved an average AUC of 0.857 and average F-measure of 0.699, while WWIC obtained a 0.835 average AUC and a 0.692 average F-measure. The P-values of the differences on AUC from WW are 5.27×10^{-9} and 0.021, respectively, and P-values on F-measures are 1.05×10^{-5} and 9.17×10^{-7}, respectively.

4. Conclusions

A kernel version of weighted canonical correlation analysis is proposed, which is implemented using a derived form of the generalized eigenvalue problem. Similar to the linear CCA and its kernelized version, this can be applied to machine learning problems for dimension reduction and feature extraction. The paper presents an application to improving the prediction quality obtained in the protein-ligand interaction problem setting. By adding bias to more similar samples, better prediction can be made which is evident on the higher AUC and F-measure values obtained. Weighting scheme on SVM based on CCA outputs were also explored and are judged to be better than classical SVM.

Even in the field of computational biology, CCA for more than two data sources has been widely used [12], [16], [17] and their usual objectives involve maximizing the sum of correlations for every pair of data sources. For future work, it could be worth exploring the extension of weighted CCA for analysis of multiple data sets in a biological setting. It could also be interesting to investigate the effectiveness of applying the proposed method to other biological problems aside from protein-ligand interaction prediction.

Acknowledgments RR acknowledges the support of the Ministry of Education, Culture, Sports, Science and Technology of Japan. TK was supported by Grant-in-Aid for Scientific Research(C) 23500373.

References

[1] Bajorath, J.: Computational analysis of ligand relationships within target families., *Curr Opin Chem Biol*, Vol. 12, No. 3, pp. 352–8 (2008).

[2] Ballester, P. J. and Mitchell, J. B.: A machine learning approach to predicting protein-ligand binding affinity with applications to molecular docking., *Bioinformatics*, Vol. 26, No. 9, pp. 1169–75 (2010).

[3] Biniashvili, T., Schreiber, E. and Kliger, Y.: Improving classical substructure-based virtual screening to handle extrapolation challenges., *J Chem Inf Model*, Vol. 52, No. 3, pp. 678–85 (2012).

[4] Bleakley, K. and Yamanishi, Y.: Supervised prediction of drug-target interactions using bipartite local models., *Bioinformatics*, Vol. 25, No. 18, pp. 2397–403 (2009).

[5] Campillos, M., Kuhn, M., Gavin, A.-C., Jensen, L. and Bork, P.: Drug target identification using side-effect similarity, *Science*, Vol. 321, pp. 263–266 (2008).

[6] Chang, C.-C. and Lin, C.-J.: LIBSVM: a library for support vector machines, *ACM Transactions on Intelligent Systems and Technology*, Vol. 2, pp. 27:1–27:27 (2011).

[7] Chen, B., Harrison, R. F., Papadatos, G., Willett, P., Wood, D. J., Lewell, X. Q., Greenidge, P. and Stiefl, N.: Evaluation of machine-learning methods for ligand-based virtual screening., *J Comput Aided Mol Des*, Vol. 21, No. 1-3, pp. 53–62 (2007).

[8] Jacob, L. and Vert, J. P.: Protein-ligand interaction prediction: an improved chemogenomics approach., *Bioinformatics*, Vol. 24, No. 19, pp. 2149–56 (2008).

[9] Klabunde, T.: Chemogenomic approaches to drug discovery: similar receptors bind similar ligands.

[10] Martin, Y., Kofron, J. and Traphagen, L.: Do structurally similar molecules have similar biological activity, *J. Med. Chem*, Vol. 45, pp. 4350–4358 (2002).

[11] Paolini, G. V., Shapland, R. H. B., van Hoorn, W. P., Mason, J. S. and Hopkins, A. L.: Global mapping of the pharmacological space, *Nat Biotechnol*, Vol. 24, pp. 805–815 (2006).

[12] Peng, Y., Zhang, D. and Zhang, J.: A New Canonical Correlation Analysis Algorithm with Local Discrimination, *Neural Process Lett*, Vol. 31, pp. 1–15 (2010).

[13] Rognan, D.: Chemogenomic approaches to rational drug design., *Br J Pharmacol*, Vol. 152, No. 1, pp. 38–52 (2007).

[14] Saito, H., Kubota, M., Roberts, R. W., Chi, Q. and Matsunami, H.: RTP family members induce functional expression of mammalian odorant receptors., *Cell*, Vol. 119, No. 5, pp. 679–91 (2004).

[15] Schuffenhauer, A., Floersheim, P., Acklin, P. and Jacoby, E.: .

[16] Tang, C. S. and Ferreira, M. A.: A gene-based test of association using canonical correlation analysis., *Bioinformatics*, Vol. 28, No. 6, pp. 845–50 (2012).

[17] Yamanishi, Y., Vert, J. P., Nakaya, A. and Kanehisa, M.: Extraction of correlated gene clusters from multiple genomic data by generalized kernel canonical correlation analysis., *Bioinformatics*, Vol. 19 Suppl 1, pp. i323–30 (2003).

[18] Yan, X. and Han, J.: gSpan: Graph-Based Substructure Pattern Mining, *Proc. 2002 Int'l Conf. Data Mining (ICDM '02)*, pp. 721–724 (2002).